Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)
Edisi. .. Volume. ... Bulan 20.. ISSN : 2089-9033

PENGEMBANGAN SISTEM REKOMENDASI APLIKASI ANDROID UNTUK ANAK DI KAKATU

Dodi Permana

Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
E-mail : do.di_permana@yahoo.com

ABSTRAK

Salah satu fitur Kakatu yang merupakan *launcher* Android adalah pembatasan aplikasi. Fitur pembatasan aplikasi dapat digunakan oleh orangtua untuk memilih aplikasi apa saja yang boleh digunakan anak. Fitur pembatasan aplikasi pada Kakatu yang merupakan parental control berbentuk launcher Android Agar pembatasan aplikasi tersebut berjalan sesuai fungsinya, orangtua harus memiliki pengetahuan tentang aplikasi apa saja yang baik digunakan oleh anak. Sistem rekomendasi dapat menjadi solusi dari masalah tersebut dengan secara otomatis menyaring dan menampilkan aplikasi apa saja yang baik digunakan oleh anak.

Knowledge-based merupakan pendekatan yang merekomendasikan suatu item berdasarkan kecocokan beberapa parameter atau fitur yang dimiliki oleh item tersebut terhadap kebutuhan pengguna. Pada domain aplikasi Android, *textual review* merupakan salah satu parameter yang dapat dipertimbangkan untuk menghasilkan rekomendasi dengan terlebih dahulu memberi label positif atau negatif dengan menggunakan klasifikasi sentimen. **S**alah satu metode klasifikasi sentimen yang baik adalah *Support Vector Machine.*

Berdasarkan analisis dan pengujian yang telah dilakukan, sistem rekomendasi aplikasi android yang dibagun telah dapat menghasilkan rekomendasi sesuai dengan syarat perekomendasian yang sebelumnya telah dianalisis dan ditetapkan. Dapat ditarik kesimpulan bahwa sistem rekomendasi aplikasi android ini layak digunakan oleh pihak Kakatu untuk memberikan rekomendasi kepada pengguna.

Kata kunci : *Kakatu, Knowledge-based, Klasifikasi Sentimen, Sistem rekomendasi, SVM*.

1. PENDAHULUAN

Kakatu adalah sebuah media *parental control* berupa *launcher* Android yang memiliki beberapa fitur utama diantaranya pembatasan aplikasi, pembatasan waktu penggunaan, dan laporan aktivitas penggunaan aplikasi. Fitur pembatasan aplikasi dapat digunakan oleh orangtua untuk memilih aplikasi apa saja yang boleh digunakan anak. Aplikasi yang sudah dibatasi diharapkan terbebas dari konten pornografi, horor, maupun kekerasan.

Agar pembatasan aplikasi tersebut berjalan sesuai fungsinya, orangtua harus memiliki pengetahuan tentang aplikasi apa saja yang baik digunakan oleh anak. Berdasarkan hasil wawancara dengan pihak Kakatu, diketahui bahwa banyaknya jumlah dan jenis aplikasi pada perangkat Android membuat pengguna kesulitan dalam mencari ataupun menentukan aplikasi yang baik. Orangtua sebagai pengguna dapat dengan tidak sengaja mengizinkan aplikasi yang tidak baik untuk digunakan oleh anak. Sistem rekomendasi merupakan sebuah model program atau teknik yang dapat menjadi solusi dari masalah tersebut dengan secara otomatis menyaring dan menampilkan aplikasi apa saja yang baik digunakan oleh anak.

Sistem rekomendasi memiliki beberapa pendekatan, salah satunya adalah *knowledge-based* [1]. *Knowledge-based* merupakan pendekatan yang merekomendasikan suatu item berdasarkan kecocokan beberapa parameter atau fitur yang dimiliki oleh item tersebut terhadap kebutuhan pengguna. Pada domain aplikasi Android, *textual review* merupakan salah satu parameter yang dapat dipertimbangkan untuk menghasilkan rekomendasi. Agar dapat digunakan oleh *knowledge-based*, *review* tersebut terlebih dahulu diberi label positif atau negatif dengan menggunakan klasifikasi sentimen. Menurut [2] salah satu metode klasifikasi sentimen yang baik adalah *Support Vector Machine*.

Berdasarkan permasalah tersebut diperlukan sebuah sistem rekomendasi aplikasi Android untuk anak pada media *parental control* Kakatu dengan menggunakkan *knowledge-based* dan klasifikasi sentimen.

1.1. Parental Control

Parental control merupakan sebuah fitur yang biasanya digunakan pada layanan televisi digital, komputer, video game, ponsel ataupun perangkat lunak. Secara umum *parental control*

terbagi ke dalam empat kategori: *content filtering*, berfungsi membatasi akses terhadap konten yang tidak layak digunakan oleh usia tertentu; *usage controls*, berfungsi untuk membatasi penggunaan perangkat dengan menentukan batas waktu penggunaan atau melarang penggunaan fitur tertentu; *computer usage management tool*, yang mewajibkan menggunakan beberapa perangkat lunak tertentu; dan monitoring, yang berfungsi untuk melacak lokasi ataupun aktifitas saat menggunakan perangkat [3].

1.2. Kakatu

Kakatu merupakan sebuah aplikasi yang dikembangkan oleh PT. Kinest Kreatif Ideata dan berada di bawah Direktur Riset dan Pengembangan. Aplikasi ini adalah cara yang aman, mudah, dan menyenangkan bagi anak dalam menggunakan *device* Android. Selain itu Kakatu dapat menjadi sebuah media *parental control* untuk mengawasi dan membatasi penggunaan handphone anak agar anak tidak sampai mengakses konten aplikasi yang tidak seharusnya.

Aplikasi ini menawarkan fungsionalitas berbeda kepada dua pengguna dengan peran berbeda, yakni orangtua dan anak. Orangtua dapat memanfaatkan fitur pembatasan aplikasi, Pembatasan panggilan telpon masuk, Pembatasan waktu penggunaan HP anak, dan Meihat riwayat penggunaan HP anak.

1.3. Knowledge-based Recommender Systems

Sistem rekomendasi dengan pendekatan yang sering digunakan (*content-based* dan *collaborative filtering*) sangat efektif jika digunakan untuk merekomendasikan *item* berdasarkan selera pengguna seperti pada kasus dengan *item* buku, film, ataupun artikel berita [1]. Ketika sistem rekomendasi bergantung kepada kebutuhan pengguna yang didefinisikan secara eksplisit, kedua pendekatan tersebut bukanlah pilihan terbaik karena bentuk atau perumusan dari kebutuhan pengguna yang didefinisikan secara eksplisit tidak sesuai dengan framework dari pendekatan *content-based* maupun *collaborative filtering*.

Knowledge-based adalah pendekatan sistem rekomendasi dengan menggunakan kebutuhan pengguna yang didefinisikan secara eksplisit dalam bentuk *recommendation rules*. Pendekatan ini terbagi kembali ke dalam dua tipe dasar yaitu *case-based* dan *constraint-based* [4]. *Case-based* menghasilkan rekomendasi dengan cara menghitung ukuran kemiripan item (menggunakan *threshold* tertentu) dengan kebutuhan yang telah ditentukan oleh pengguna. *Constraint-based* menghasilkan rekomendasi dengan mencari sekumpulan item yang dapat memenuhi *recommendation rules*. Secara umum *constraint-based* dapat dipresentasikan sebagai *constraint satisfaction problem* (CSP) dan dapat diselesaikan dengan menggunakan CSP *solver* ataupun menggunakan *conjunctive query* yang dieksekusi dan diselesaikan oleh *database engine*. *Conjunctive query* memandang permasalah rekomendasi sebagai data *filtering* [4]. *Conjunctive query* adalah *query database* yang berisi sekumpulan kriteria seleksi yang saling terhubung, *query* tersebut didefinisikan sebagai:

$$\sigma_{[criteria]}(P) \quad (1)$$

Dimana σ merepresentasikan operator seleksi, *criteria* adalah *recommendation rules* yang berlaku, dan P adalah tabel dari kumpulan produk yang tersedia.

1.4. Support Vector Machine

Klasifikasi sentimen adalah proses mengkategorisasikan sebuah dokumen yang tidak berlabel ke dalam kategori yang telah ditentukan. Secara umum permasalahan dalam klasifikasi sentimen dapat digambarkan dengan: jika diberikan satu set dokumen D, fungsi yang disebut classifier akan mengklasifikasi setiap dokumen d ϵ D kepada salah satu dari kelas positif atau negatif. Positif memiliki arti bahwa dokumen d mengekspresikan opini positif dan negatif memiliki arti bahwa dokumen d mengkspresikan opini negatif [2].

Pada klasifikasi sentimen data yang digunakan berasal dari dokumen atau teks yang tidak terstruktur. Oleh karena itu, dibutuhkan *text preprocessin*g yang berfungsi untuk mengubah bentuk data yang semula tidak terstruktur menjadi data yang terstruktur.

1.5. Web Scraping

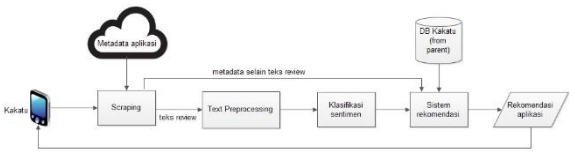
Web scraping merupakan sekumpulan teknik untuk mendapatkan informasi dari sebuah *website* secara otomatis tanpa harus menyalinnya secara manual. Tujuan dari *web scraping* adalah untuk mencari informasi tertentu, mengekstraksi informasi tersebut, kemudian menghimpunnya dalam halaman *web* yang baru. Beberapa *web scraper* memfokuskan dirinya untuk mengubah data yang tidak terstruktur dan menyimpannya ke dalam data yang terstruktur.

Aplikasi *web scraping* (umumnya disebut juga dengan agen yang cerdas dan otomatis) berfokus hanya pada mendapatkan data dengan cara pengambilan dan ekstraksi serta dapat melibatkan kumpulan data dengan perbedaan ukuran yang signifikan [5].

2. ISI PENELITIAN

2.1. Analisis Arsitektur Sistem

Analisis arsitektur sistem digunakan untuk menganalisis perangkat lunak yang dibangun dalam bentuk gambaran sistem secara umum. Arsitektur sistem dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur sistem rekomendasi

Berikut adalah pemaparan dari arsitektur sistem tersebut:

1. Pengguna menggunakan fungsionalitas Pembatasan Aplikasi pada aplikasi Kakatu. Aplikasi kemudian mengirimkan id setiap aplikasi yang ter-*install* pada *device* melalui jaringan internet dan API.
2. Sistem melakukan *scraping* terhadap Android Market dengan menggunakan id setiap aplikasi untuk mendapatkan *metadata* dari aplikasi tersebut.
3. *Metadata* aplikasi yang berupa teks *review* akan melalui tahap *preprocessing* yang terdiri dari *case folding*, *stopword removal*, *covert emoticon*, *convert negation*, *tokenization*, *stemming*, dan *term weighting*. Tahap ini bertujuan agar teks *review* tersebut terbebas dari *noise* dan dapat dikenali oleh *classifier* yang berada pada tahap klasifikasi sentimen.
4. Tahap klasifikasi sentimen akan melakukan klasifikasi dengan memberikan label positif atau negatif kepada setiap *review* sesuai dengan sentimen atau opini yang terkandung dalam review tersebut. *Review* yang telah memiliki label akan menjadi salah satu parameter penentu pada tahapan sistem rekomendasi.
5. Sistem rekomendasi berbasis *knowledge* akan menghasilkan rekomendasi berdasarkan *requirement* yang didefinisikan secara eksplisit. *Requirement* tersebut berupa ketentuan metadata yang harus dipenuhi oleh sebuah aplikasi yang diantaranya adalah review yang sudah memiliki label positif ataupun negatif.
6. Selanjutnya, hasil rekomendasi berupa sekumpulan aplikasi yang telah melalui *filtering* dikirimkan kembali kepada pengguna melalui jaringan internet dan API.

2.2. Analisis Data Masukan

Data masukan yang digunakan adalah kumpulan aplikasi ID yang ter-*install* pada *device* pengguna Kakatu, riwayat perizinan setiap aplikasi, serta *metadata* aplikasi yang didapat dari Android Market. Data tersebut nantinya akan dijadikan sebagai parameter untuk menghasilkan rekomendasi. Berikut adalah pemaparan dari setiap data masukan tersebut:

1. Aplikasi ID

Aplikasi ID adalah sebuah variabel unik yang dimiliki oleh setiap aplikasi. Pada penelitian ini aplikasi ID tersebut berupa package name yang didapat dari pengguna Kakatu.

2. Rating pengguna

Android Market seperti Google Play menggunakan rating dengan sistem angka berskala 1 sampai 5.

3. Rating konten

Rating konten merupakan rating aplikasi menurut tingkat kedewasaan. Rating ini akan menyarankan kesesuaian usia dari pengguna terhadap konten yang terkandung pada aplikasi atau game. Rating konten di Google Play diantaranya: Rating 3+, Rating 7+, Rating 12+, Rating 16+, dan Rating 18+.

4. Review Pengguna

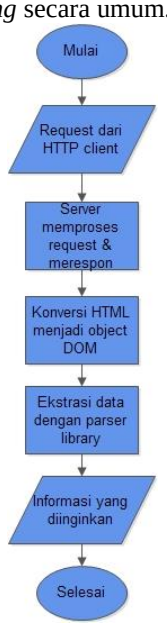
Review pengguna merupakan ulasan pengguna terhadap sebuah aplikasi dalam bentuk teks. Review ini dapat memiliki sentimen/opini negatif ataupun positif.

5. Riwayat perizinan aplikasi

Data/parameter ini mempresentasikan berapa kali sebuah aplikasi telah diizinkan orangtua untuk digunakan oleh anak mereka. Data ini berbentuk nilai presentase yang didapat dari API yang telah disediakan oleh pihak Kakatu.

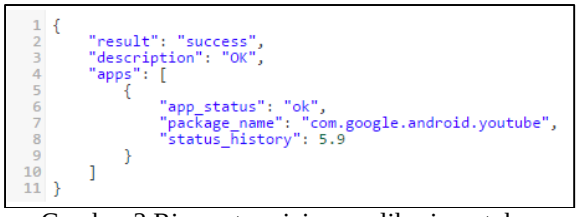
2.3. Analisis Metode Scraping dan Pemanfaatan API

Scraping digunakan untuk mendapatkan *metadata* aplikasi yang bersangkutan dari Android Market. Gambar 2 menjelaskan diagram alir dari proses *web scraping* secara umum.



Gambar 2 Diagram alir *web scraping*

Sedangkan untuk mendapatkan riwayat perizinan aplikasi digunakan sebuah API yang telah tersedia pada server Kakatu dengan mengakses https://www.kakatu.web.id/status-history.json menggunakan method HTTP POST. Sebagai contoh, output riwayat perizinan dari aplikasi Youtube dalam bentuk format JSON adalah sebagai berikut:



```
{
    "result": "success",
    "description": "OK",
    "apps": [
        {
            "app_status": "ok",
            "package_name": "com.google.android.youtube",
            "status_history": 5.9
        }
    ]
}
```

Gambar 3 Riwayat perizinan aplikasi youtube

2.4. Analisis Metode Klasifikasi Sentimen

Agar *metadata* yang berbentuk teks *review* dapat diklasifikasikan dengan baik diperlukan tahap *preprocessing*. Pada tahap ini data *review* yang digunakan untuk data *training* dan data *testing* direduksi, dibersihkan dari *noise*, dan dibuat menjadi terstruktur. Berikut ini adalah tahapan-tahapan preprocessing yang dilakukan dalam penelitian ini:

1. *Case folding*

Case folding merupakan tahap merubah huruf kapital (*uppercase*) menjadi huruf kecil (*lowercase*). Hal ini dilakukan agar semua huruf pada *review* menjadi seragam.

2. *Stopword removal*

Beberapa kata yang tidak mengandung sentimen dimasukkan ke dalam daftar *stopword*. Kata ini biasanya berupa kata ganti orang, kata ganti penghubung, paranomial petunjuk dan lain sebagainya. Pada tahap ini kata pada *review* yang juga terdapat dalam *stopword* akan dihilangkan.

3. *Convert emoticon*

Convert emoticon adalah tahap mengkonversikan *emoticon* ke dalam *string* yang sesuai dengan ekspresi *emoticon* itu sendiri. Tahap ini dilakukan karena *emoticon* menjadi salah satu cara pengguna mengekspresikan pendapatnya. Hal tersebut dapat mempunyai pengaruh terhadap pengklasifikasian sentimen.

4. *Convert negation*

Convert negation merupakan tahap konversi kata-kata negasi yang terdapat pada suatu *review* aplikasi, kata negasi mempunyai pengaruh dalam merubah nilai sentimen pada suatu *review*. Jika terdapat kata negasi pada suatu *review* maka kata tersebut akan disatukan dengan kata setelahnya.

5. *Tokenization*

Tahap *tokenization* dilakukan dengan cara memisahkan kata berdasarkan spasi yang terletak diantara dua kata. Tahapan ini juga menghilangkan simbol atau tanda baca yang terdapat dalam *review* aplikasi.

6. *Stemming*

Pada penelitian ini tahap *stemming* yang digunakan untuk mendapatkan setiap kata dasar dari setiap kata menggunakan algoritma Porter.

7. *Term Weighting*

Tahap *preprocessing* terakhir dalam penelitian ini adalah *term weighting*. Data *review* yang telah melalui tahap *case folding*, *stopword removal*, *convert emoticon*, *convert negation*, *tokenization*, dan *stemming* akan di-*index* dan diberi bobot.

Setelah melalui tahap *preprocessing*, data *review* diklasifikasikan ke dalam dua kelas berbeda yaitu kelas negatif dan kelas positif dengan menggunakan *Support Vector Machine* (SVM). Terdapat dua tahapan utama pada algoritma SVM yakni tahap pelatihan dan tahap pengklasifikasian. Tahap pelatihan berfungsi untuk membuat sebuah model yang nantinya akan digunakan oleh algoritma SVM untuk mengklasifikasi data yang sebelumnya belum pernah diketahui. Dalam tahap pelatihan dibutuhkan dokumen latih dengan kelas positif dan negatif dimana penilaian negatif dan positif tersebut dilakukan secara manual. Pada penelitian ini digunakan 500 *review* pengguna beberapa aplikasi android yang terbagi menjadi 250 review positif dan 250 review negatif. Setelah model didapatkan, proses pengklasifikasian *review* baru dapat dilakukan.

2.5. Analisis Metode Sistem Rekomendasi

Pada penelitian ini metode sistem rekomendasi yang digunakan adalah *constraint-based recommendation*. Secara umum metode ini sangat bergantung kepada informasi karakteristik dari item yang akan direkomendasikan. Untuk menghasilkan rekomendasi, Kakatu memiliki beberapa kriteria karakteristik yang harus dipenuhi oleh setiap aplikasi seperti:

1. Rating konten dari aplikasi adalah rating 3+ dan rating 7+.
2. Rating pengguna aplikasi sekurang-kurangnya adalah 3.5.
3. Aplikasi tersebut setidaknya pernah diizinkan lebih dari 5% Orangtua untuk digunakan oleh anak mereka.
4. Aplikasi harus memiliki *review* pengguna yang bersifat positif melebihi 50% dari total keseluruhan *review*.

Rating konten, rating pengguna, dan *review* pengguna dari aplikasi yang bersangkutan didapat dari tahap *scraping*. Untuk mengetahui banyaknya *review* pengguna yang bersifat positif, maka dilakukan tahapan *text preprocessing* serta klasifikasi sentimen sehingga *review* yang bersangkutan memiliki label positif atau negatif. Jumlah persentase untuk *review* yang bersifat positif dari sebuah aplikasi dihitung dengan:

$$Persentase\ review\ positif = \frac{jumlah\ review\ positif}{total\ review} \times 100$$

Sedangkan riwayat perizinan aplikasi didapat dari *database* sistem berjalan dengan menggunakan API yang disediakan oleh pihak Kakatu.

Sebagai contoh, Tabel 1 menggambarkan daftar aplikasi yang ter-*install* pada perangkat Android pengguna dan telah melalui tahap *scraping*, *text preprocessing*, dan klasifikasi sentimen.

Tabel 1 Daftar aplikasi yang ter-install pada perangkat Android pengguna

Aplikasi	Rating Konten	Rating pengguna	Riwayat perizinan (%)	*Review* positif dari pengguna (%)
YouTube	Rated for 12+	4.1	15	60
Instagram	Rated for 12+	4.5	0	70
Teach Kids Arabic Alphabet	Rated for 3+	4.0	50	80
Playtime With Dora	Rated for 7+	3.9	60	53
SimSimi	Rated for 16+	4.1	3	73
Pepi Bath	Rated for 3+	3.0	21	68
Chrome Browser - Google	Rated for 3+	4.2	6	88
Barbara's Second Child Birth	Rated for 7+	3.8	39	55

Permasalahan *constraint-based recommendation* yang diselesaikan oleh *conjunctive query* dilakukan dengan cara mengeksekusi *query database* dengan sebuah set kriteria yang saling terhubung secara konjungtif. Berdasarkan kriteria yang telah ditetapkan dan daftar aplikasi yang tersedia, rekomendasi dapat dihasilkan dengan menggunakan persamaan (1) dimana:

P : {Youtube, Instagram, Teach Kids Arabic Alphabet, Playtime With Dora, SimSimi, Pepi Bath, Chrome Browser – Google, Barbara's Second Child Birth}

criteria : rating_konten = rating 3+ or rating_konten = rating 7+, rating_pengguna > 3.5, riwayat_perizinan > 20, review_positif > 50

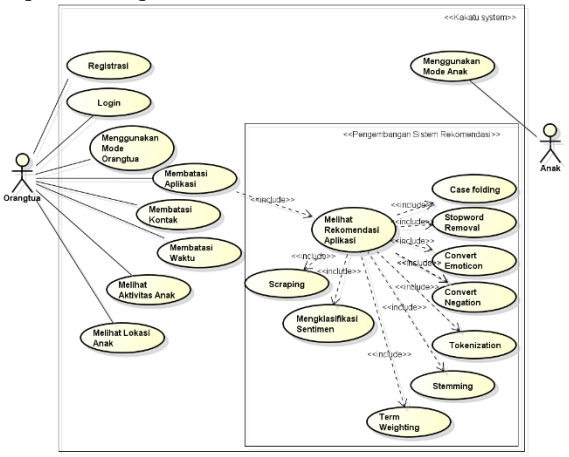
Dengan demikian:

$\sigma_{[criteria]}$ (P) = {Teach Kids Arabic Alphabet,Playtime With Dora,Barbara's Second Child Birth}

Dapat disimpulkan bahwa dari sekumpulan aplikasi yang terdapat dalam tabel tersebut Teach Kids Arabic Alphabet, Playtime With Dora, dan Barbara's Second Child Birth adalah aplikasi yang akan direkomendasikan oleh sistem.

2.6. Analisis Kebutuhan Fungsional

Use case diagram dari sistem yang diusulkan dapat dilihat pada Gambar 4.



Gambar 4 *Use case diagram* sistem yang diusulkan

2.7. Implementasi dan Pengujian

Bagian ini memaparkan bagian implementasi dan pengujian dari system yang dibangun.

2.7.1. Lingkungan Implementasi

Spesifikasi perangkat keras (*hardware*) yang digunakan dalam proses perancangan dan implementasi dapat dilihat pada Tabel 2.

Tabel 2 Spesifikasi perangkat keras

No	Hardware	Spesifikasi
1	Processor	Intel(R) Core(TM) i3 CPU M 350 @ 2.27Ghz (4 CPUs), ~2.3GHz
2	Harddisk	340 GB
3	Memory	4GB
4	VGA	2438 MB
5	Monitor	14 inch
6	Mouse	Wireless Logitech
7	Keyboard	QWERTY
9	Koneksi Internet	Ya

Spesifikasi perangkat lunak (*software*) yang digunakan dalam proses perancangan dan implementasi dapat dilihat pada Tabel 3.

Tabel 3 Spesifikasi perangkat keras

No	Software	Spesifikasi
1	Sistem Operasi	Windows 8.1 Professional
2	Bahasa Pemrograman	PHP, JSON
3	DBMS	MySql
4	Browser	Google Chrome Ver 43.0.2357.132 m (64-bit)
5	Text Editor	Sublime Text 3
6	Web Server	XAMPP ver 3.2.1

2.7.2. Implementasi Data

Pada penelitian ini perancangan data diimplementasikan dengan menggunakan MySQL. Bentuk implementasi tersebut adalah sebagai berikut:

1. Pembuatan database

Query:

```
CREATE DATABASE sistem_rekomendasi;
```

2. Pembuatan tabel aplikasi

Query:

```
CREATE TABLE IF NOT EXISTS
`aplikasi` (
`id_aplikasi` int(11) NOT NULL
AUTO_INCREMENT,
`nama_aplikasi` varchar(100) NOT
NULL,
`package_name` varchar(100) NOT
NULL,
`rating_pennguna` double NOT NULL,
`rating_konten` varchar(50) NOT
NULL,
`riwayat_perizinan` float NOT NULL,
`status` enum('y','t') DEFAULT NULL,
PRIMARY KEY ('id_aplikasi')
) ENGINE=InnoDB;
```

3. Pembuatan tabel review_pengguna

Query:

```
CREATE TABLE IF NOT EXISTS
`review_pengguna` (
`id_review` int(11) NOT NULL
AUTO_INCREMENT,
`id_aplikasi` int(11) NOT NULL,
`review_teks` varchar(500) NOT NULL,
`hasil_preprocessing` varchar(500)
DEFAULT NULL,
`kelas` varchar(15) DEFAULT NULL,
PRIMARY KEY ('id_review'),
FOREIGN KEY ('id_aplikasi')
REFERENCES 'aplikasi'
('id_aplikasi')
) ENGINE=InnoDB;
```

4. Pembuatan tabel training_svm

Query:

```
CREATE TABLE IF NOT EXISTS
`data_training` (
`id_training` int(11) NOT NULL
AUTO_INCREMENT,
`review_teks` varchar(300) NOT
NULL,
`kelas` varchar(15) NOT NULL,
PRIMARY KEY ('id_training')
) ENGINE=InnoDB;
```

5. Pembuatan tabel stoplist

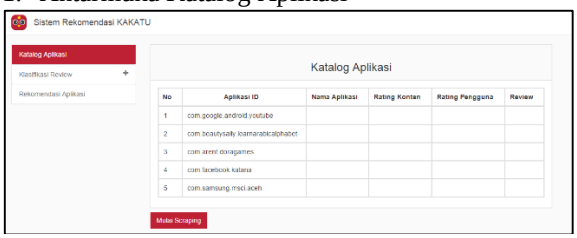
Query:

```
CREATE TABLE IF NOT EXISTS
`stoplist` (
`id_stopword` int(11) NOT NULL
AUTO_INCREMENT,
`stopword` varchar(50) NOT NULL,
PRIMARY KEY ('id_stopword')
) ENGINE=InnoDB;
```

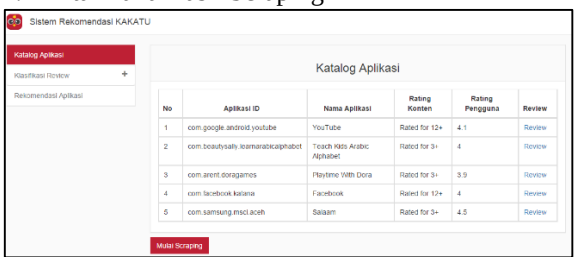
2.7.3. Implementasi Antarmuka

Hasil implementasi antarmuka dari perancangan tersebut adalah sebagai berikut:

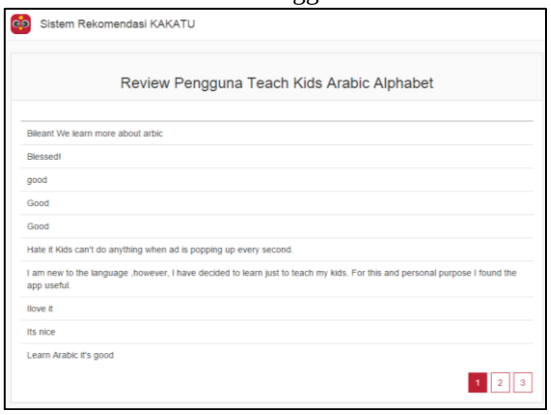
1. Antarmuka Katalog Aplikasi



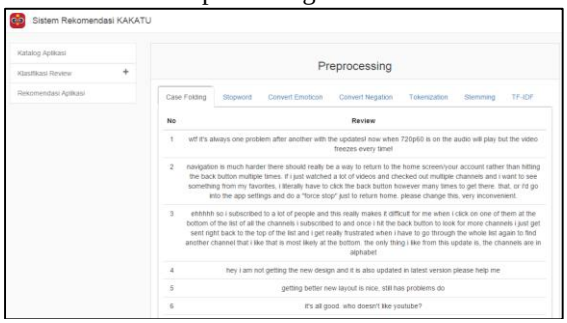
2. Antarmuka Hasil Scraping



3. Antarmuka Review Pengguna



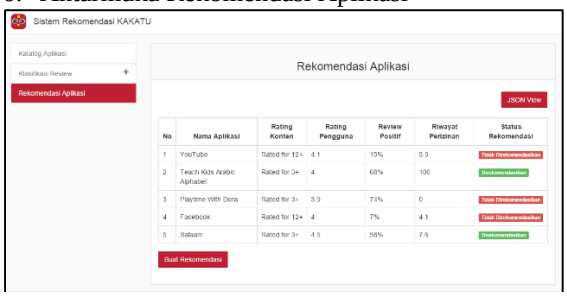
4. Antarmuka Preprocessing



5. Antarmuka Klasifikasi Review



6. Antarmuka Rekomendasi Aplikasi



2.7.4. Pengujian

Hasil dan evaluasi dari pengujian sistem yang dilakukan yakni pengujian fungsionalitas, pengujian *acceptance*, pengujian sample testing, dan pengujian pengukuran akurasi adalah sebagai berikut:

1. Pengujian Fungsionalitas

Pengujian fungsionalitas berfokus pada persyaratan fungsional dari sebuah sistem yang dibangun dan menemukan kesalahan program. Berikut adalah hasil dari beberapa kelas uji:

a. Pengujian Scraping

Hasil pengujian untuk fungsionalitas Scraping dapat dilihat pada Tabel 4.

Tabel 4 Hasil pengujian fungsionalitas scraping

Kondisi Eksekusi	Keluaran yang diharapkan	Pengamatan	Kesimpulan
Menekan tombol Scraping	Setelah menekan tombol Scraping, sistem menampilkan nama aplikasi, rating konten, rating pengguna, dan review dari *package name* yang bersangkutan.	Sistem menampilkan nama aplikasi, rating konten, rating pengguna, dan review dari *package name* yang bersangkutan.	Diterima

b. Pengujian Preprocessing

Hasil pengujian untuk fungsionalitas Preprocessing dapat dilihat pada Tabel 5.

Tabel 5 Hasil pengujian fungsionalitas preprocessing

Kondisi Eksekusi	Keluaran yang diharapkan	Pengamatan	Kesimpulan
Menekan tombol Preprocessing	Setelah menekan tombol Preprocessing, sistem menampilkan review dari *package name* yang bersangkutan yang telah melalui tahap preprocessing.	Sistem menampilkan review dari *package name* yang bersangkutan yang telah melalui tahap preprocessing.	Diterima

c. Pengujian Klasifikasi Review

Hasil pengujian untuk fungsionalitas Klasifikasi Review dapat dilihat pada Tabel 6.

Tabel 6 Hasil pengujian fungsionalitas klasifikasi review

Kondisi Eksekusi	Keluaran yang diharapkan	Pengamatan	Kesimpulan
Menekan tombol Klasifikasi	Setelah menekan tombol Klasifikasi, sistem menampilkan review dari *package name* yang bersangkutan yang telah memiliki kelas positif atau negatif.	Sistem menampilkan review dari *package name* yang bersangkutan yang telah memiliki kelas positif atau negatif.	Diterima

d. Pengujian Pembuatan Rekomendasi

Hasil pengujian untuk fungsionalitas Pembuatan Rekomendasi dapat dilihat pada Tabel 7.

Tabel 7 Hasil pengujian fungsionalitas pembuatan rekomendasi

Kondisi Eksekusi	Keluaran yang diharapkan	Pengamatan	Kesimp-ulan
Menekan tombol Buat Rekomendasi	Setelah menekan tombol Buat Rekomendasi, sistem menampilkan nama aplikasi, rating konten, rating pengguna, jumlah review positif dalam bentuk persen, riwayat perizinan, dan status rekomendasi dari *package name* yang bersangkutan.	Sistem menampilkan nama aplikasi, rating konten, rating pengguna, jumlah review positif dalam bentuk persen, riwayat perizinan, dan status rekomendasi dari *package name* yang bersangkutan.	Diterima

Berdasarkan hasil pengujian fungsional yang dilakukan maka dapat disimpulkan bahwa sistem yang dibangun menunjukan output yang diharapakan.

2. Pengujian *Acceptance*

Pengujian *acceptance* yang dilakukan adalah dengan melakukan wawancara langsung kepada narasumber yakni CEO dan CTO di PT. Kinest Kreatif Ideata. Kesimpulan jawaban dari hasil wawancara dapat dilihat pada Tabel 8.

Tabel 8 Kesimpulan hasil wawancara

No	Pertanyaan	Kesimpulan Hasil Wawancara
1	Apakah sistem yang dibangun sudah dapat memberikan rekomendasi aplikasi android yang baik untuk anak?	Sudah dapat memberikan rekomendasi
2	Apakah penentuan rekomendasi aplikasi android dengan menggunakan pendekatan *knowledge-based recommender system* (membuat rekomendasi berdasarkan ketentuan yang didefinisikan secara eksplisit) sudah sesuai dengan visi dari fitur pembatasan aplikasi yang dimiliki oleh Kakatu?	Sudah sesuai
3	Apakah rekomendasi aplikasi yang dihasilkan oleh sistem tersebut valid atau sesuai dengan kriteria yang telah ditetapkan sebelumnya?	Valid
4	Secara keseluruhan, apakah sistem yang dibangun dapat diimplementasikan terhadap fitur pembatasan aplikasi yang sudah berjalan?	Ya

Berdasarkan hasil pengujian *acceptance* pada Tabel 8, dapat disimpulkan bahwa sistem yang dibangun sudah dapat memberikan rekomendasi aplikasi android dengan benar.

3. Pengujian *Sample Testing*

Pengujian dilakukan dengan cara membandingkan *output* dari sistem terhadap data *sample* yang telah disiapkan. Tahap ini dilakukan untuk menguji ada tidaknya kesalahan dalam mengimplementasikan logika dari metode yang diterapkan. *Data sample* yang digunakan adalah beberapa data *review* yang telah memiliki label (positif/negatif) dan beberapa *package name* dari aplikasi yang telah memiliki status rekomendasi (direkomendasikan/tidak direkomendasikan). Data review yang dijadikan sample testing dapat dilihat pada Tabel 9.

Tabel 9 Data *review* yang dijadikan sample

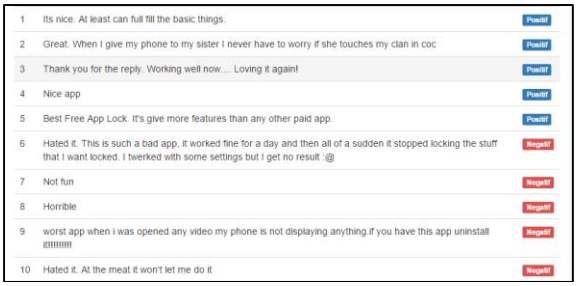
No	Review	Label
1	Its nice At least can full fill the basic things.	Positif
2	Great When I give my phone to my sister I never have to worry if she touches my clan in coc	Positif
3	Thank you for the reply. Working well now.... Loving it again!	Positif
4	Nice app	Positif
5	Best Free App Lock It's give more features than any other paid app.	Positif
6	Hated it This is such a bad app, it worked fine for a day and then all of a sudden it stopped locking the stuff that I want locked. I twerked with some settings but I get no result	Negatif
7	Not fun	Negatif
8	Horrible	Negatif
9	worst app when i was opened any video my phone is not displaying anything.if you have this app uninstall it!!!!!!!!!	Negatif
10	Hated it. At the meat it won't let me do it	Negatif

Data *package name* yang dijadikan sebagai sample testing dapat dilihat pada Tabel 10.

Tabel 10 Data *package name* dari aplikasi yang dijadikan *sample*

No	Package Name	Status Rekomendasi
1	com.google.android.youtube	Tidak Direkomendasikan
2	com.beautysally.learnarabicalphabet	Direkomendasikan
3	com.arent.doragames	Tidak Direkomendasikan
4	com.facebook.katana	Tidak Direkomendasikan
5	com.samsung.msci.aceh	Direkomendasikan

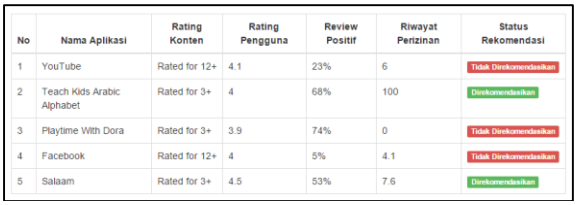
Hasil dari pengujian *sample testing* terhadap *review* berlabel dapat dilihat pada Gambar 5.



1	Its nice. At least can full fill the basic things.	Positif
2	Great. When I give my phone to my sister I never have to worry if she touches my clan in coc	Positif
3	Thank you for the reply. Working well now.... Loving it again!	Positif
4	Nice app	Positif
5	Best Free App Lock. It's give more features than any other paid app.	Positif
6	Hated it. This is such a bad app, it worked fine for a day and then all of a sudden it stopped locking the stuff that I want locked. I twerked with some settings but I get no result :@	Negatif
7	Not fun	Negatif
8	Horrible	Negatif
9	worst app when i was opened any video my phone is not displaying anything.if you have this app uninstall it!!!!!!!!!	Negatif
10	Hated it. At the meat it won't let me do it	Negatif

Gambar 5 Hasil klasifikasi oleh sistem

Hasil dari pengujian data *package name* dengan status rekomendasi dapat dilihat pada Gambar 6.



No	Nama Aplikasi	Rating Konten	Rating Pengguna	Review Positif	Riwayat Perizinan	Status Rekomendasi
1	YouTube	Rated for 12+	4.1	23%	6	Tidak Direkomendasikan
2	Teach Kids Arabic Alphabet	Rated for 3+	4	68%	100	Direkomendasikan
3	Playtime With Dora	Rated for 3+	3.9	74%	0	Tidak Direkomendasikan
4	Facebook	Rated for 12+	4	5%	4.1	Tidak Direkomendasikan
5	Salaam	Rated for 3+	4.5	53%	7.6	Direkomendasikan

Gambar 6 Hasil rekomendasi oleh sistem

Berdasarkan hasil pengujian *sample testing* yang telah dilakukan, menujukan bahwa *output* dari sistem tidak berbeda dengan data yang dijadikan *sample testing*.

4. Pengujian Pengukuran Akurasi

Pengujian pengukuran akurasi dilakukan menggunakan metode *k-fold cross validation* dengan nilai k sebanyak 10 *fold*. Pengujian ini bertujuan untuk menguji stabilitas akurasi jika diuji dengan data latih dan data uji yang berbeda. Pengujian dilakukan pada pengklasifikasian teks *review* yang menggunakan metode *support vector machine*. Data *review* yang digunakan sebanyak 500 data yang dibagi menjadi 10 subset yang masing-masing subset berjumlah 50 data. Data *review* tersebut telah memiliki label positif atau negatif. Skenario uji akurasi dengan metode 10-*fold cross validation* dapat dilihat pada Tabel 10.

Tabel 10 Skenario Uji 10-*fold cross validation*

Percobaan	Data Review	*Subset*
Fold 1	Data Training	$S_2, S_3, S_4, S_5, S_6, S_7 S_8, S_9, S_{10}$
	Data Testing	S_1
Fold 2	Data Training	$S_1, S_3, S_4, S_5, S_6, S_7 S_8, S_9, S_{10}$
	Data Testing	S_2
Fold 3	Data Training	$S_1,S_2, S_4, S_5, S_6, S_7 S_8, S_9, S_{10}$
	Data Testing	S_3
Fold 4	Data Training	$S_1,S_2, S_3, S_5, S_6, S_7 S_8, S_9, S_{10}$
	Data Testing	S_4
Fold 5	Data Training	$S_1,S_2, S_3, S_4, S_6, S_7 S_8, S_9, S_{10}$
	Data Testing	S_5
Fold 6	Data Training	$S_1,S_2, S_3, S_4, S_5, S_7, S_8, S_9, S_{10}$
	Data Testing	S_6
Fold 7	Data Training	$S_1,S_2, S_3, S_4, S_5, S_6, S_8, S_9, S_{10}$
	Data Testing	S_7
Fold 8	Data Training	$S_1,S_2, S_3, S_4, S_5, S_6, S_7S_9, S_{10}$
	Data Testing	S_8
Fold 9	Data Training	$S_1,S_2, S_3, S_4, S_5, S_6, S_7 , S_{10}$
	Data Testing	S_9
Fold 10	Data Training	$S_1,S_2, S_3, S_4, S_5, S_6, S_7 S_8, S_9$
	Data Testing	S_{10}

Pada skenario uji dengan jumlah 10-*fold cross validation* dan iterasi yang dilakukan sebanyak 10 kali iterasi. Iterasi pertama pada *fold* pertama menggunakan 450 data gabungan dari subset ke-2 sampai subset ke-9 sisanya yaitu 50 data yang digunakan sebagai data *testing* pada subset pertama, proses *training* dan proses *testing* dilakukan sampai iterasi ke-10. Detail hasil dari akurasi pengujian skenario pada masing-masing *fold*-nya dapat dilihat pada Tabel 11.

Tabel 11 Hasil uji akurasi klasifikasi review dengan SVM

Fold	Jumlah Data Training	Jumlah Data Testing	
1	450	50	
2	450	50	
3	450	50	
4	450	50	
5	450	50	
6	450	50	
7	450	50	
8	450	50	
9	450	50	
10	450	50	
Akurasi			66%

Pada pengujian menggunakan metode 10-*fold cross validation* menunjukkan bahwa metode SVM dan data yang digunakan sebanyak 500 *review* memiliki tingkat akurasi yang cukup memuaskan yakni 66%.

3. PENUTUP

3.1. Kesimpulan

Berdasarkan analisis dan pengujian yang telah dilakukan, diperoleh kesimpulan bahwa sistem rekomendasi aplikasi android yang dibagun telah dapat menghasilkan rekomendasi aplikasi yang baik

untuk anak sesuai dengan syarat perekomendasian yang sebelumnya telah dianalisis dan ditetapkan dengan melibatkan pihak Kakatu.

3.2. Saran

Sistem yang dibangun tidak membahas *packaging product* secara mendalam, oleh karena itu adapun saran yang dapat diberikan dalam pengembangan selanjutnya yakni meningkatkan waktu komputasi dari sistem dengan memperhatikan penggunaan *resource* dan kecepatan akses data dari *server*.

DAFTAR PUSTAKA

[1] F. Ricci, L. Rokach, B. Shapira and P. B. Kantor, Recommender Systems Handbook, New York: Springer, 2011.

[2] V. K. Singh, M. Mukherjee and G. K. Mehta, "Combining Collaborative Filtering and Sentiment Classification for Improved Movie Recommendations," *Multi-disciplinary Trends in Artificial Intelligence*, pp. 38-50, 2011.

[3] D. Morley, Understanding Computers: Today and Tomorrow, 2009 Update, Content Technologies, Inc., 2014

[4] M. Fitri, "Perancangan Sistem Temu Balik Informasi Dengan Metode Pembobotan Kombinasi Tf-Idf Untuk Pencarian Dokumen Berbahasa Indonesia," 2012.

[5] M. Turland, php|architect's Guide to Web Scraping with PHP, Toronto: Marco Tabini, 2010.